Missão Prática Nível 1 – Mundo 3

Nome: Lucas Garcia Resende

Matrícula: 202308135163

Polo Petrópolis - RJ
Iniciando o Caminho pelo Java – Turma 9001 – 2024.3

Objetivo da Prática

O objetivo do primeiro procedimento é a criação de um sistema simples para o cadastro e recuperação de dados de uma pessoa física ou jurídica. A prática involve a criação de uma classe Pessoa que serve como o modelo básico para as outras duas classificações, exigindo apenas uma identificação numérica e um nome para a inicialização de um objeto da classe. A classe oferece métodos básicos: setters, getters e um método de exibição das informações da pessoa, sendo esse último método modificado posteriormente pelas subclasses de Pessoa com as informações especificas da classe fílha.

A partir da classe Pessoa, é criada duas novas subclasses, PessoaFisica e PessoaJuridica, a primeira inicializando com, além da identificação e nome, um valor numérico representando a idade da pessoa física e um texto contendo o CPF, como novos setters e getters e modificando o método polimórfico exibir. A classe Pessoa implementa a interface Serializable, permitindo que ela e todas suas subclasses sejam armazenadas e recuperadas em arquivos binários.

Por fim, é criado duas novas classes, PessoaFisicaRepo e PessoaJuridicaRepo, com o objetivo de criar um vetor contendo vários objetos da sua respectiva subclasse de Pessoa, com o objetivo principal de serem armazenados e recuperados. Ambas contém métodos para modificar, excluir ou adicionar pessoas armazenadas no vetor, retornar um ou todos os objetos e, principalmente um método para armazenar os objetos em um arquivo binário e um método para recuprar os objetos em um arquivo criado para persistir as informações. As operações de armazenamentos e recuparação são testados pela classe principal do projeto e exibidas no terminal.

Código Fonte:

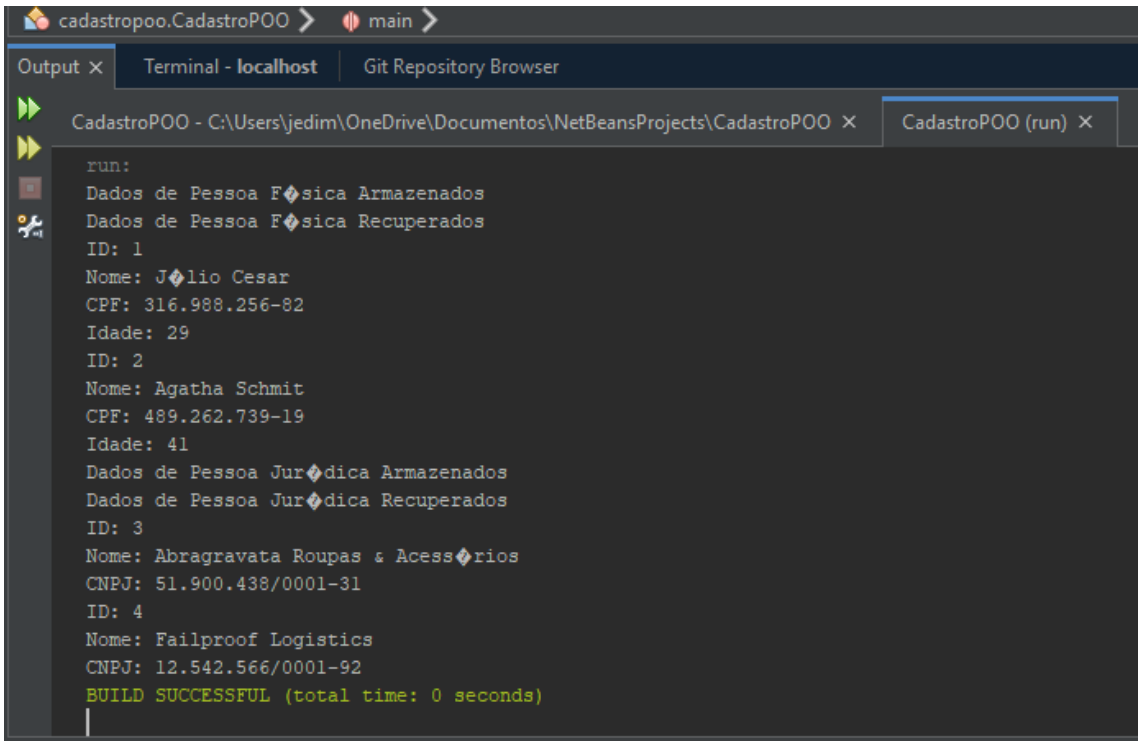
Classe Principal:

https://github.com/LucasGarciaResende/CadastroPOO/blob/9db833897c927d4eef7c6437a334ac4690083fd3/src/cadastropoo/CadastroPOO.java

Classes Modelo:

https://github.com/LucasGarciaResende/CadastroPOO/tree/master/src/model

Resultado da execução do código:



Análise e Conclusão:

A) Quais as vantagens e desvantagens do uso de herança?

A principal vantagem de utilizar herança é a reutilização de códigos, permitindo o programador a economizar várias linhas de códigos ao permitir subclasses herdar as funcionalidades de outras classes. No caso do programa da prática, as classes PessoaFisica e PessoaJuridica herdam funcionalidades da classe parente Pessoa.

Uma das disvantagens que a herança apresenta é tornar o código menos legível em um sistema mais complexo, pois uma classe pode herdar diversas funcionalidades que não aparentes no código da mesma, outro problema que o programa pode apresentar ao ser utilizado herança é que modificar uma classe parente, consequentemente, modificara suas classes filhas, o que pode causar comportamentos inexperados de uma ou mais subclasses se uma mudança na classe parente não é feita com cuidado.

B) Por que a interface Serializable é necessária ao efetuar persistência em arquivos binários?

A interface Serializable permite a JVM reconhecer que a classe pode ter objetos serializados de forma segura, sem ela, objetos podem não ser convertidos em um arquivo binário corretamente.

C) Como o paradigma funcional é utilizado pela API stream no Java?

A API de Streams do Java permite aplicar conceitos do paradigma funcional, apesar do Java ser uma linguagem orientada a objetos. Funcionalidades como expressões lambda e funções puras são incorparadas a partir do Streams do Java.

D) Quando trabalhamos com Java, qual padrão de desenvolvimento é adotado na persistência de dados em arquivos?

O padrão utilizado é o DAO, ou Data Access Object, esse padrão permite fácil acesso e compatibilidade com diversos sistemas de armazenamento de dados

Missão Prática Nível 1 – Mundo 3

Nome: Lucas Garcia Resende

Matrícula: 202308135163

Polo Petrópolis - RJ
Iniciando o Caminho pelo Java – Turma 9001 – 2024.3

Objetivo da Prática

O objetivo do segundo procedimento é criar uma interface via terminal para o uso das funcionalidades do programa. A partir de um do-while loop, o programa pode receber via input do usuário a opção de inserir, alterar, excluir ou obter uma pessoa física ou jurídica no seu respectivo objeto de repositório e também de mostrar todos os objetos armazenados. Por fim, permite persistir ou recuperar os objetos PessoaFisica ou Juridica via arquivos binários.

Código Fonte:

Classe Principal:

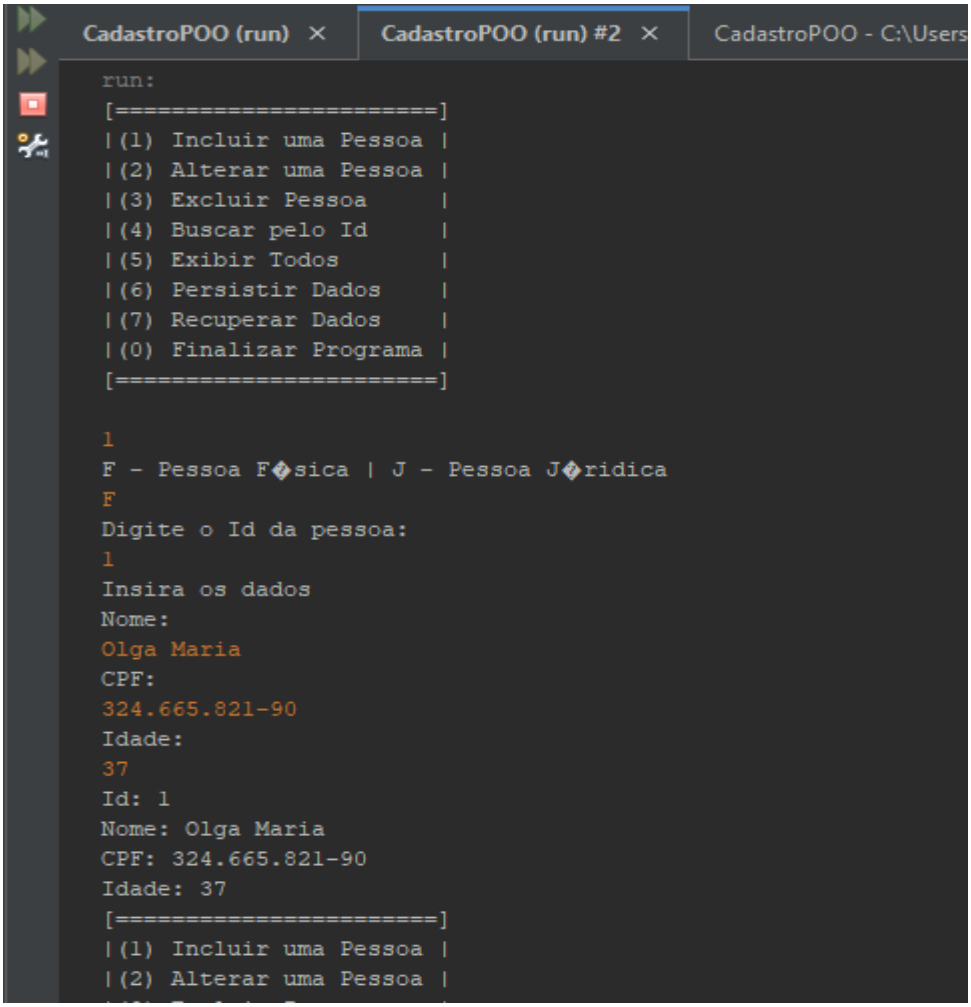
https://github.com/LucasGarciaResende/CadastroPOO/blob/master/src/cadastropoo/CadastroPOO.java

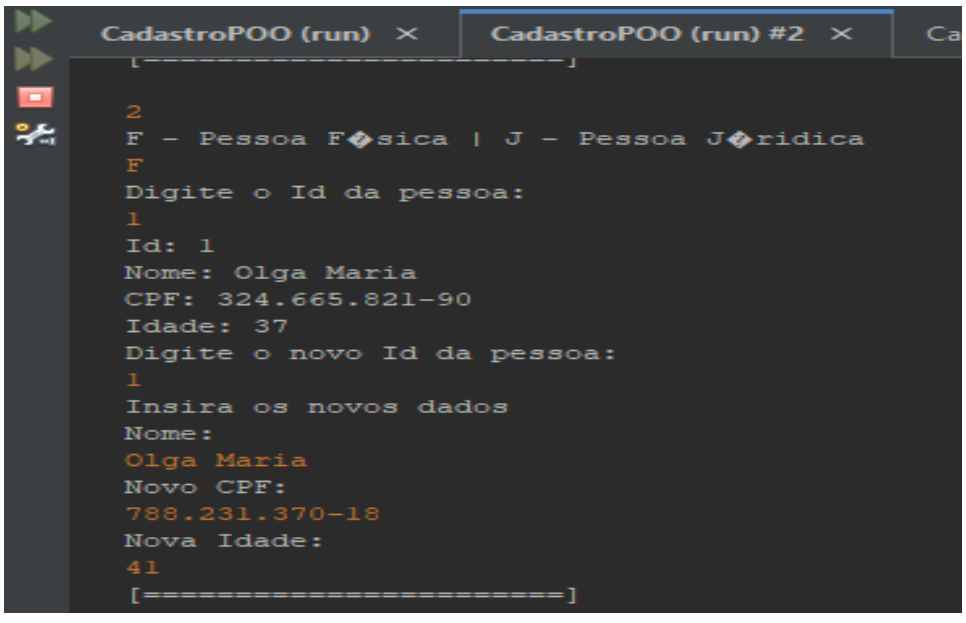
Classes Modelo:

https://github.com/LucasGarciaResende/CadastroPOO/tree/master/src/model

Estácio

Resultado da execução do código:






```
F - Pessoa F♦sica | J - Pessoa J♦ridica
F
Digite o Id da pessoa a ser excluida:
1
[======================]
```

```
4
F - Pessoa F♦sica | J - Pessoa J♦ridica
F
Digite o Id da pessoa a ser excluida:
2
Id: 2
Nome: Igor Amaral
CPF: 126.990.875-35
Idade: 23
[======================]
|(1) Incluir uma Pessoa |
```

```
5
F - Pessoa F♦sica | J - Pessoa J♦ridica
F
Id: 2
Nome: Igor Amaral
CPF: 126.990.875-35
Idade: 23
Id: 1
Nome: Lara Oliveira
CPF: 821.367.010-72
Idade: 31
[======================]
|(1) Incluir uma Pessoa |
```

```
Digite o nome do arquivo de pessoa f♦sica para persistir
pessoa_fisica02
Digite o nome do arquivo de pessoa jur♦dica para persistir
pessoa_juridica02
Dados de Pessoa F♦sica Armazenados
Dados de Pessoa Jur♦dica Armazenados
Arquivos salvos!
[======================]
```

```
7
Digite o nome do arquivo de pessoa fisica para recuperar
pessoa_fisica02
Digite o nome do arquivo de pessoa jur♦dica para recuperar
pessoa_juridica02
Dados de Pessoa F♦sica Recuperados
Dados de Pessoa Jur♦dica Recuperados
Arquivos recuperados!
[======================]
|(1) Incluir uma Pessoa |
```

Análise e Conclusão:

A) O que são elementos estáticos e qual o motivo para o método main adotar esse modificador?

Elementos estáticos de uma classe são elementos, tanto atributos quando métodos, que existem na classe sem a necessidade de um objeto da classe ser instanciado, isso que significa que idenpedente de como for iniciado um objeto, esses elementos permaneceram exatamente iguais em todas as instâncias. Esses elementos são declarados pelo modificar “static”.

B) Para que serve a classe Scanner?

A classe Scanner serve para receber inputs na execução do programa. Métodos da classe como nextLine() permitem o código receber um bloco de texto em um programa dinâmico.

C) Como o uso de classes de repositório impactou na organização do código?

Teve um bom impacto na organização do código, isolando as funcionalidades do repositório e armazenamento dos objetos em duas classes permitiu o código ser muito mais conciso e legível.